

REMARKS

Claims 1-24 are pending in the present application. Applicants respectfully traverse the Examiner's rejections of claims 1-24 in view of the reasons set forth herein.

In the Office Action, claims 1-24 were rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by *Draves* (U.S. Patent No. 5,802,590). Applicants respectfully traverse the Examiner's § 102 rejections. An anticipating reference by definition must disclose every limitation of the rejected claim in the same relationship to one another as set forth in the claim. For restricting the execution of security sensitive instructions, independent claims 1, 9, and 17 set forth, among other things, associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by a processor and obtaining a second security ID associated with the software code running on the processor.

However, *Draves* does not teach or remotely suggest restricting the execution of security sensitive instructions by associating a first security identification (ID) with instructions and obtaining a second security ID associated with a software code (different than the instruction(s)). Instead, *Draves* pairs two sets of handles with keys for a single or same item, i.e., a computer system resource shared between two different authorized processes (a server and a client process). By using the two sets of handles with keys for the shared resource, *Draves* ensures that two different authorized processes can access that shared resource. In this manner, *Draves* does not use two security IDs associated with two different items (the requested instruction(s) and the software code) for restricting the execution of security sensitive instructions. Based on the above indicated legal standard, it is respectfully submitted that *Draves* fails to anticipate independent claims 1, 9, and 17.

Further, independent claims 1, 9, and 17 set forth, among other things, requesting to execute at least one of the plurality of instructions or set of instructions by the software code running on the processor. Claims 1, 9, and 17 also set forth executing the requested instruction or set of instructions providing that the second security ID matches the first security ID. In an exemplary embodiment of the instant invention, the Applicants' Specification describes that for restricting the execution of security sensitive instructions by the software code running on the processor 305, the processor 305 determines whether the security ID associated with the code running thereon matches the security ID associated with the particular instruction that the code is attempting to execute. If there is a mismatch between the security ID associated with the code running on the processor and the security ID associated with the particular instruction, the processor 305 denies execution of the security sensitive instruction by the software code running thereon at block 545. If there is a match between the security ID associated with the code running on the processor 305 and the security ID associated with the security sensitive instruction, the processor 305 executes the security sensitive instruction at block 550. See Applicants' Specification on page 14, lines 2-11.

Rather than requesting to execute at least one instruction by the software code running on the processor and executing the requested instruction providing that the second security ID matches the first security ID, in *Draves* access to the resource is requested by a requestor; such as a calling process (e.g., a server process) and access to the resource is granted when the stored key matches to the passed key for that resource. Accordingly, *Draves* does not use two security IDs associated with two different items for restricting the execution of security sensitive instructions by executing the requested instruction providing that two security IDs associated

with two different items match. Therefore, *Draves* fails to disclose or suggest all the limitations in the method of claim 1.

As understood, *Draves* is directed to granting only authorized processes a secure access to a shared computer system resource. The Examiner asserts that *Draves* teaches in Figures 3, reference numeral labeled "handle/key" and Figure 1 reference numeral "110/120", Column 3, line 63-64 that for each multiplicity/plurality of processes a handle/key pair is associated. Thus, according to the Examiner, *Draves* teaches use of two security IDs associated with two different items. However, *Draves* is completely silent with regard to any associating of a first security identification (ID) with each of one or more instructions that are to be executed by a processor and obtaining a second security ID associated with a software code, as set forth in claim 1.

In contrast, *Draves* teaches that when a process wishes to access the allocated resource, it simply passes the handle/key pair associated with a shared computer system resource to the kernel. The kernel examines the resource entry indexed by the passed handle to determine whether the passed key is equal to the key in the indexed resource entry. In this way, through the use of handle/key pairs, *Draves* provides a system which ensures that only authorized processes are able to access resources. The kernel allows a process access to a resource only when the passed key matches the key for the resource that is stored in the resource entry. See *Draves*, Column 3, lines 63-67. Additionally, the use of handle/key pairs also allows for compatibility with processes that are designed to use only keys. See *Draves*, Column 4, lines 35-36.

More specifically, the main memory 220 contains a client process 222, a server process 224, and a kernel 226. The handle/key pair is sent to a process. The process accesses the

resources by passing handle/key pairs to the kernel 226. The kernel 226 compares the passed key with a key that is stored in the resource entry referenced by the passed handle. When the stored key and the passed key match, the process is allowed to access the resource. That is, the server process 302 sends a resource allocation request to the kernel 304. The kernel then returns the handle/key pair 312 to the server process. The server process 302 then passes the handle/key pair 313 to the client process 314 with which the server process desires to share the resource. See *Draves*, Column 4, lines 53-62, Column 5, lines 32-36, and the Abstract.

Accordingly, Applicants respectfully submit that, in *Draves*, the handle/key pairs for the shared resource are passed by the server process 302 and are not associated to the client process 222 the way a first security identification (ID) is associated with each of the requested instruction(s) to be executed by a software code and a second security ID is associated with the software code for restricting the execution of the requested instruction(s) by the software code. Instead of requesting to execute at least one instruction by the software code running on the processor and executing the requested instruction, in *Draves*, the server process 302 sends a resource allocation request to the kernel 304 for sharing the resource with the client process 314. For at least the aforementioned reasons, Applicants respectfully submit that the present invention is not anticipated by *Draves* and request that the Examiner's rejections of claims 1-24 under 35 U.S.C. 102(b) be withdrawn.

Claims 4-6, 12-14 and 20-22 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over *Draves* in view of *Krueger* et al. (U.S. Patent No. 4,962,533). Reconsideration of the present application in view of the reasons set forth herein is respectfully requested.

Applicants submit that claims 4-6, 12-14 and 20-22 are not rendered obvious over *Draves* in view of *Krueger*. To establish a *prima facie* case of obviousness, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (CCPA 1974). The Examiner recognizes that *Draves* fails to teach or suggest classifying at least one instruction or set of instructions from a plurality of instructions that are to be executed by a processor as being security sensitive. The Examiner relies upon *Krueger* to describe these claim limitations. However, *Krueger* does not remedy the fundamental deficiencies of *Draves* discussed above.

The cited references also fail to provide any suggestion or motivation for modifying the prior art to arrive at Applicants' claimed invention. To the contrary, *Krueger* teaches away from classifying instructions as being security sensitive. For example, in Column 2, lines 47-48 and lines 53-56, *Krueger* does not check classification of an instruction accessing a word in the memory. Instead, *Krueger* is directed to controlling user access to data within a computer system. The computer system classifies data (not an instruction or instructions(s)) only at the level which is needed to provide a security technique for a computer system in which all data retains its classification, and in which no data is overclassified. In a computer system every word in the memory has a corresponding label. This label indicates the security classification, and compartments if any, of that word of data. Each time a word is accessed by any instruction, its classification is checked to see if access is allowed. Any attempt to improperly access any word within the computer system's memory generates a security violation and prohibits further execution of the currently running process. See *Krueger*, Column 2, lines 1. 33-56. It is by now well established that teaching away by the prior art constitutes *prima facie* evidence that the claimed invention is not obvious. See, *inter alia*, *In re Fine*, 5 U.S.P.Q.2d (BNA) 1596, 1599


(Fed. Cir. 1988); *In re Nielson*, 2 U.S.P.Q.2d (BNA) 1525, 1528 (Fed. Cir. 1987); *In re Hedges*, 228 U.S.P.Q. (BNA) 685, 687 (Fed. Cir. 1986).

For at least the aforementioned reasons, Applicants respectfully submit that the present invention is not obvious over the cited references, either alone or in combination. Applicants request that the Examiner's rejections of claims 4-6, 12-14 and 20-22 under 35 U.S.C. 103(a) be withdrawn.

For the aforementioned reasons, it is respectfully submitted that all claims pending in the present application are in condition for allowance. The Examiner is invited to contact the undersigned at (713) 934-4089 with any questions, comments or suggestions relating to the referenced patent application.

Respectfully submitted,

Date: 07/14/05



Sanjeev K. Singh, Ph.D.
Limited Recognition No. L0220
Williams Morgan & Amerson, P.C.
10333 Richmond Avenue, Suite 1100
Houston, TX 77042
(713) 934-7000
(713) 934-7011 (Fax)

AGENT FOR APPLICANTS